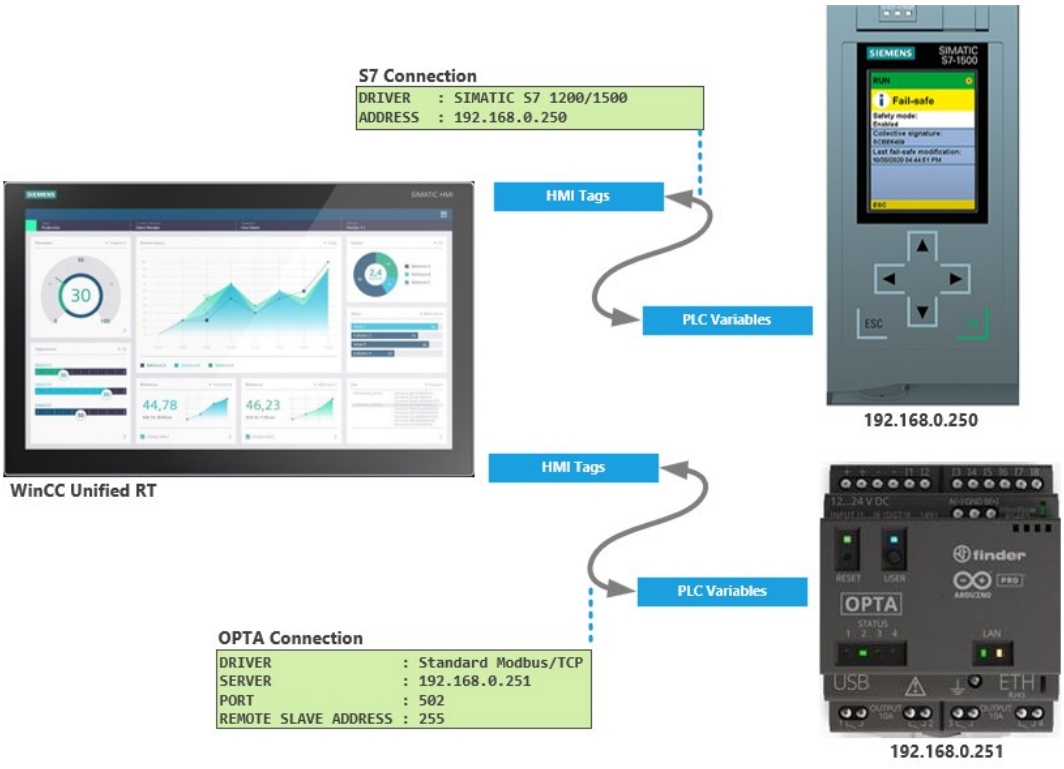
Interfacing Siemens WinCC Unified RT with Arduino OPTA

(in an existing Siemens network)

- Application Note -



Davide Nardella

WinCC Unified is a very powerful and flexible HMI platform, intended to replace the well-known WinCC Advanced/Professional in the Siemens ecosystem.

Unlike its predecessors, and in line with the best competitor systems, it is completely based on Web technology (HTML5, SVG graphics, etc.).

It allows you to scale towards highly elaborate customized applications, thanks to the ability to manage scripts in JavaScript (not a proprietary language therefore) and to create customized controls starting from open-source libraries available online.

The name Unified was born from the desire to unify applications through a single tool, starting from a simple station HMI, up to complex plant Scada.

Arduino OPTA is a small PLC, I have written several times about it (in this article and in this other one), it is a system that I really like because it combines a very low price with very interesting features.

It is born programmable relay and is also referred to as such by the manufacturer, however, it is much more like a micro-PLC, in fact, programmable relays generally use different tools than those of the larger families; they have many limitations that makes them suitable for simplified uses.

In this case we are dealing with a full compliant IEC61131-3 system that also allows for the integration of C++ code directly into the programming environment.

It is a PLC that I would call PRO/STEM, it is suitable for small professional realizations (let's not forget the market segment in which it is placed) but it is also an excellent tool for school-age or pre-professional PLC learning.

The dual nature of this PLC is evident when we go to program it; in fact, we can use both Arduino IDE, in C++, the tool of choice for Makers, and Arduino PLC IDE for a more professional use.

The purpose of this tutorial is to integrate Arduino OPTA into an existing Siemens HMI network, that is, to create a page through which we can interact with the OPTA Inputs and Outputs and view/modify its internal variables.

Let's start by saying that interfacing WinCC with OPTA is not an activity for hackers, OPTA allows communication in Modbus/TCP and WinCC manages multiple connections with mixed networks, among which there is Modbus/TCP.

So, they are two objects compatible by design.

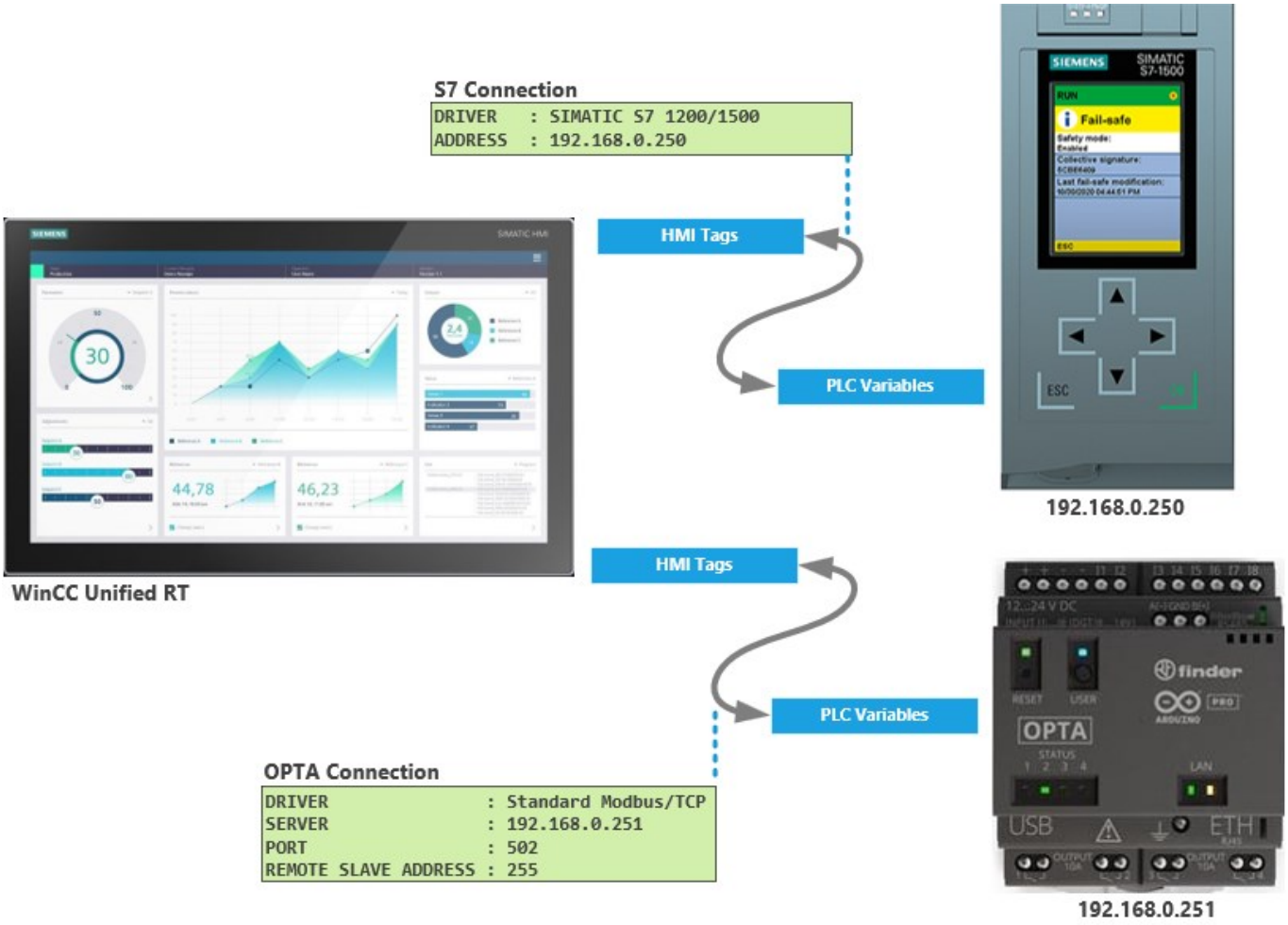
However, proceeding in a haphazard manner by changing many parameters at the same time, in addition to not offering us a logical thread, risks transforming a very simple activity into hours of fruitless attempts.

The purpose of this tutorial, therefore, is to understand how to proceed, in a simple and orderly way, to create a mixed architecture in a few minutes.

We will not see how to connect WinCC to a PLC S71200/1500, the prerequisite, therefore, is to already have a basic TIA Portal WinCC/PLC project working.

We will see how to prepare a WinCC HMI system to host a connection to OPTA and we will create together a small program for Arduino OPTA with all the variable management.

System architecture



- The HMI runtime exchanges variables (called Tags) with a PLC through a “connection” characterized by the protocol with its parameters, and the physical IP address of the PLC.

- In addition to the type (integer, real, etc.) and the connection, each runtime tag is associated with the logical address of the variable within the PLC.

- WinCC can communicate with multiple PLCs of different brands at the same time. This is achieved by managing multiple different connections at the same time.

Phase 1: Preparing Arduino OPTA

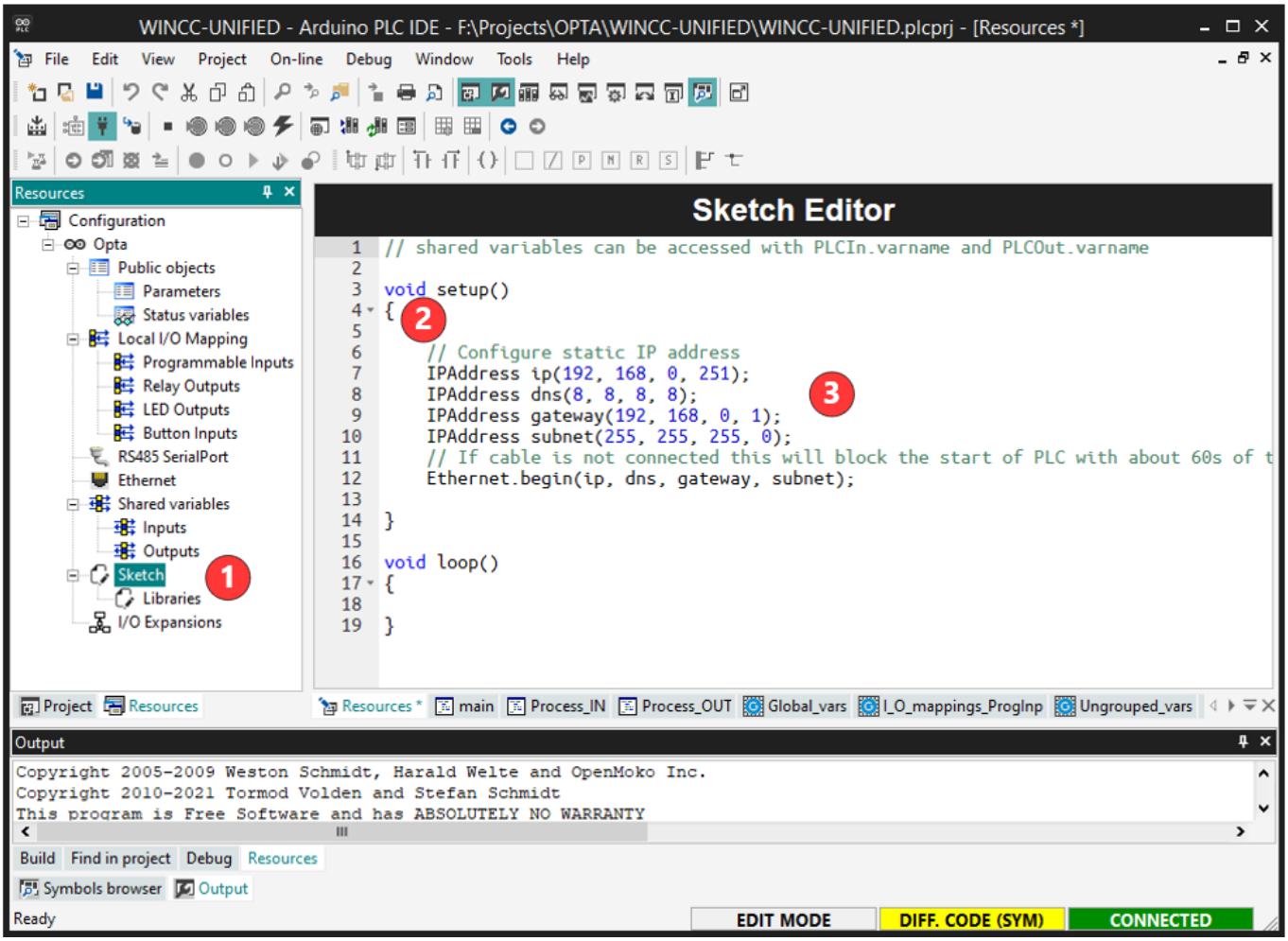
To prepare Arduino OPTA, these steps are needed.

1. Enabling the Ethernet communication
2. Creating the HMI exchange variables.
3. Hardware resources mapping (Inputs, relays, leds, etc.) into local variables.
4. Writing two small programs for preparing variables (*).
5. Writing the main program.

(*)

The cost of an HMI runtime license depends on the number of Tags used. An optimization, that is worth learning right away, is to exchange Boolean variables for bits within a word.
In this way, to exchange 16 bits, we will need only one Word Tag instead of 16 Bool Tags. Extracting or inserting bits in a Word, as we will see, is a very simple operation thanks to the possibility of addressing the individual bits directly in the variable name.

OPTA.1 – Enabling the Ethernet communication



Create a new OPTA Project with Arduino PLC IDE.

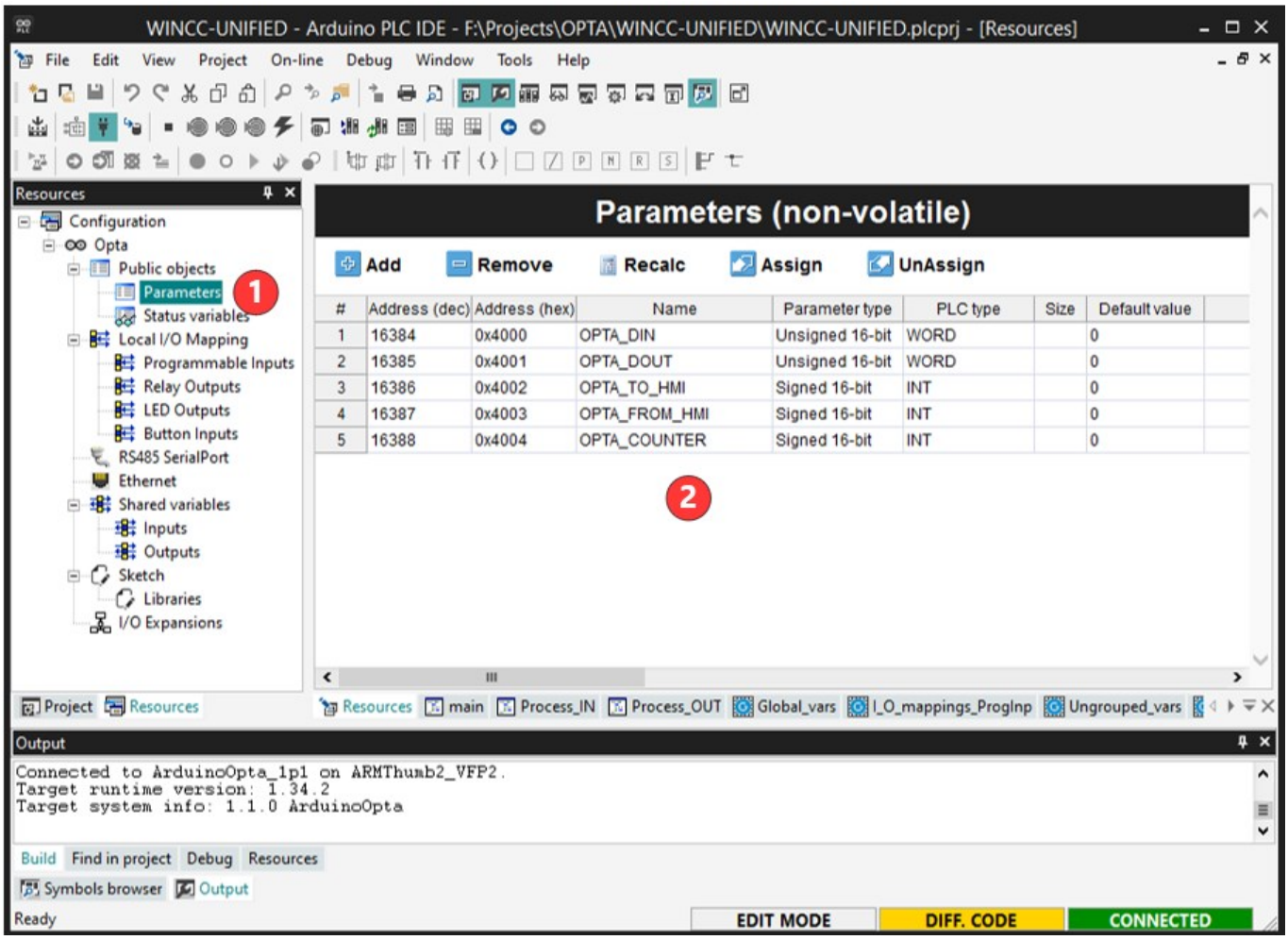
1. Select the **Sketch** item in **Resources** tab.

2. Delete the comment delimiters (/* and */) to enable the network setup.

3. Insert the OPTA IP and gateway (if any) address, which **must be compatible** with the HMI station.

OPTA.2 - Creating the HMI exchange variables

Let's create the variables that will be 1:1 with the HMI Tags.



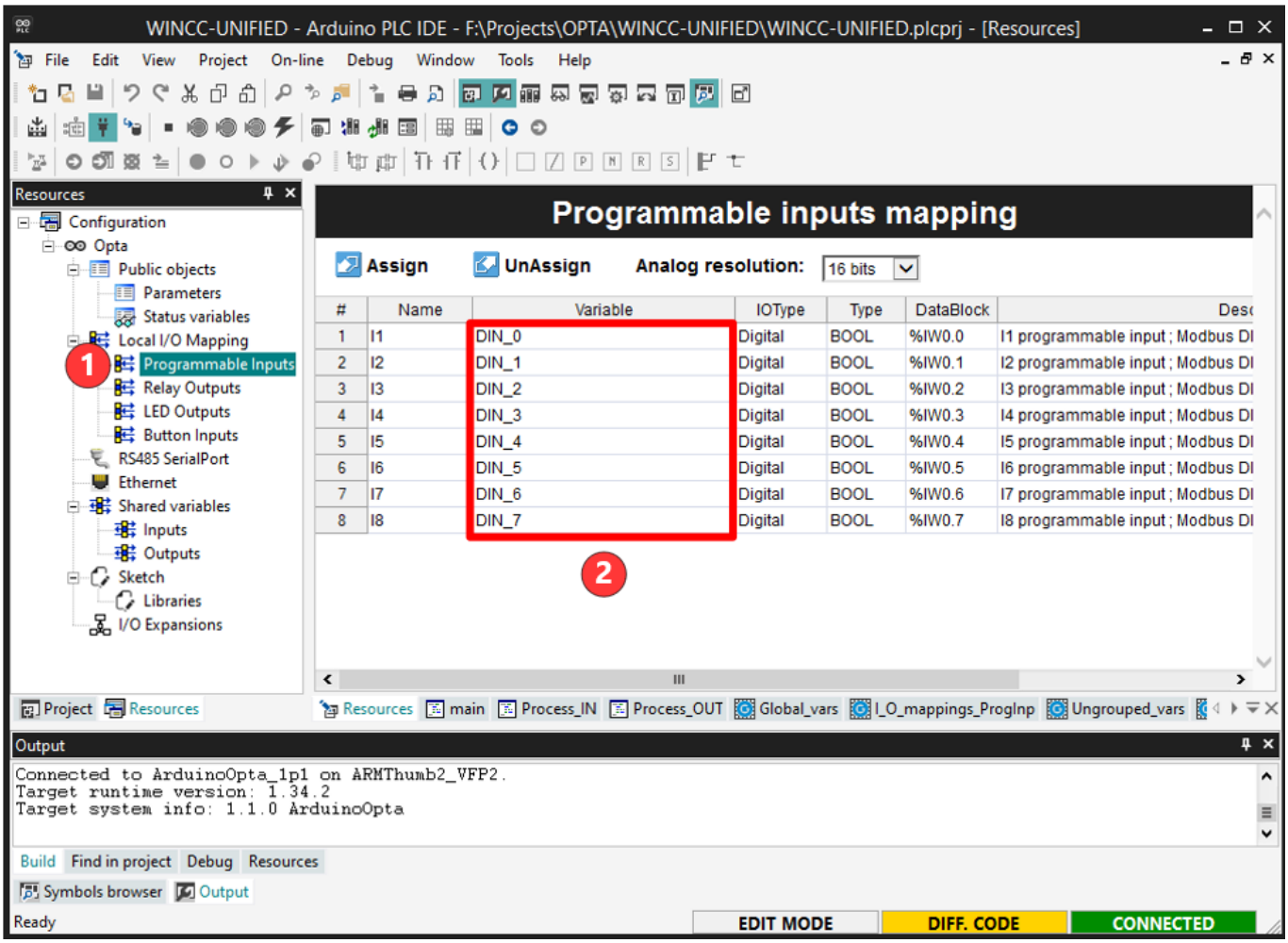
1. Select the **Parameters** item in **Resources** Tab.

2. Add the parameters as in figure, leaving unchanged the proposed address; do modify Parameter type e PLC type.

Suggestion

> *Take a screenshot of this, it will be useful when you need to insert tags into the HMI.*

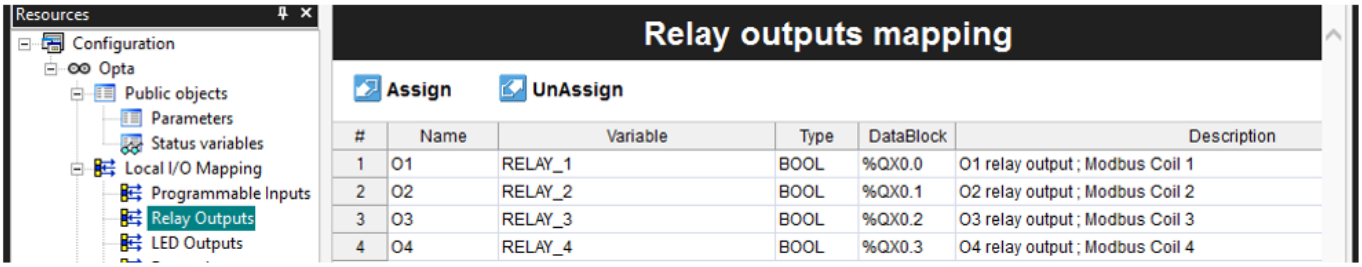
OPTA.3 – Hardware resources mapping

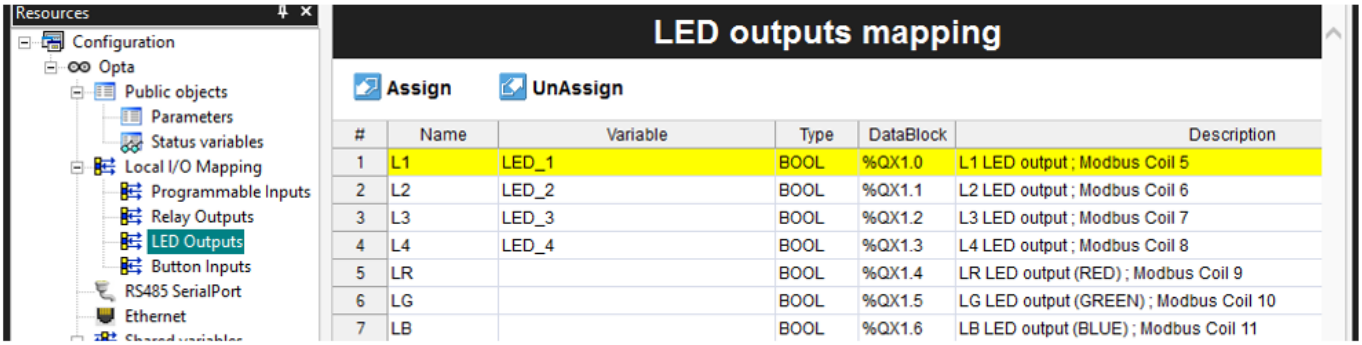


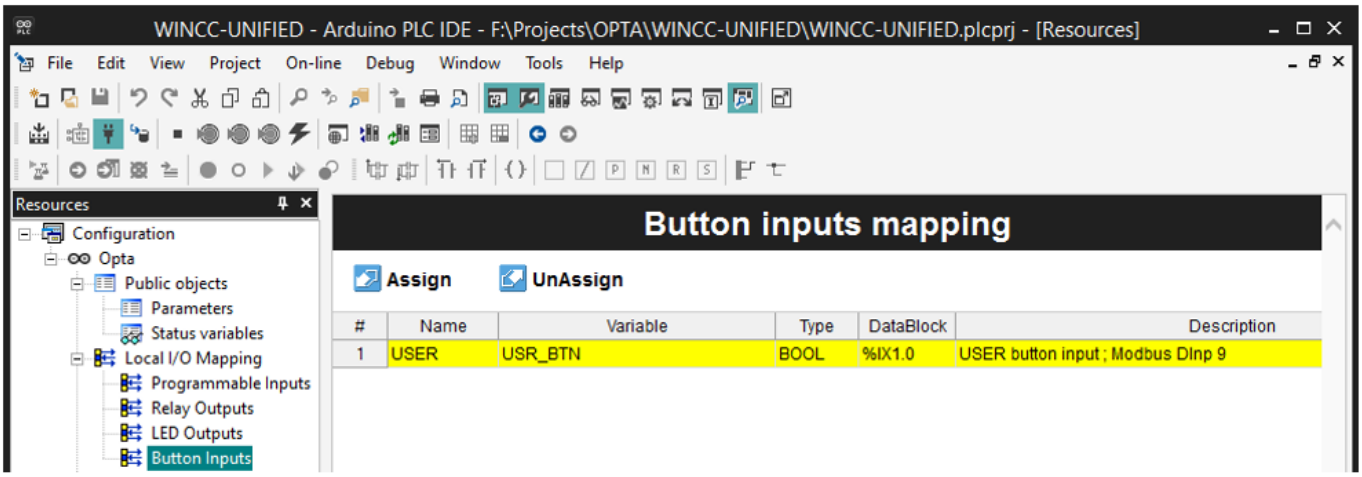
1. Select the **Programmable inputs** item in **Local I/O Mapping** group**.**

2. Add the names, as in figure, leaving unchanged the remaining values.

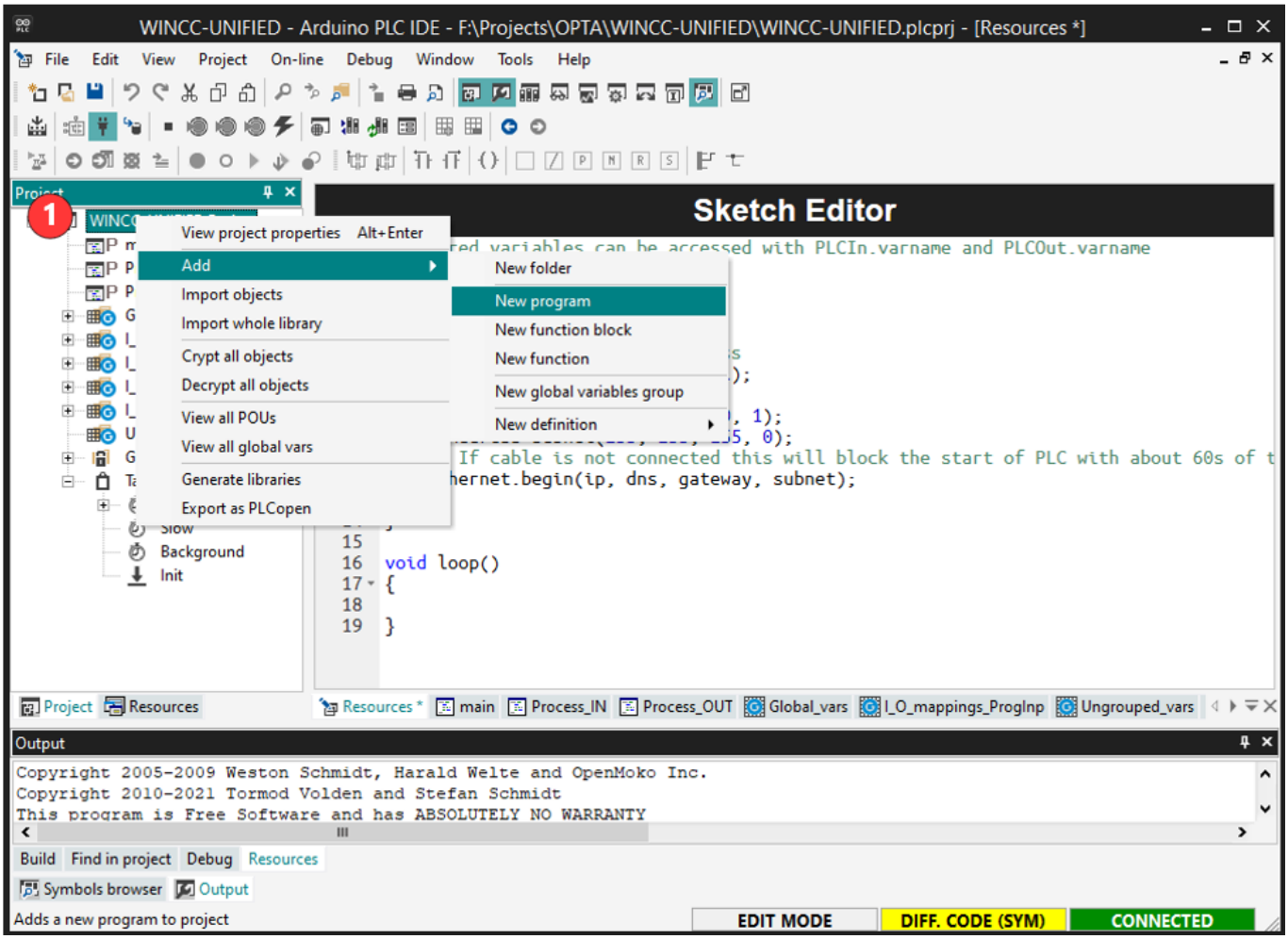
3. Do the same for **Relay**, **LED** e **Button**





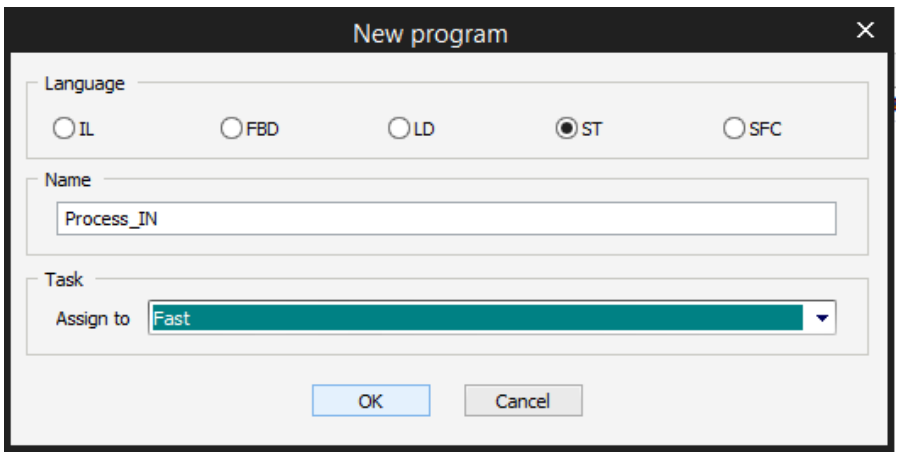


OPTA.4 – Preparing the interface programs

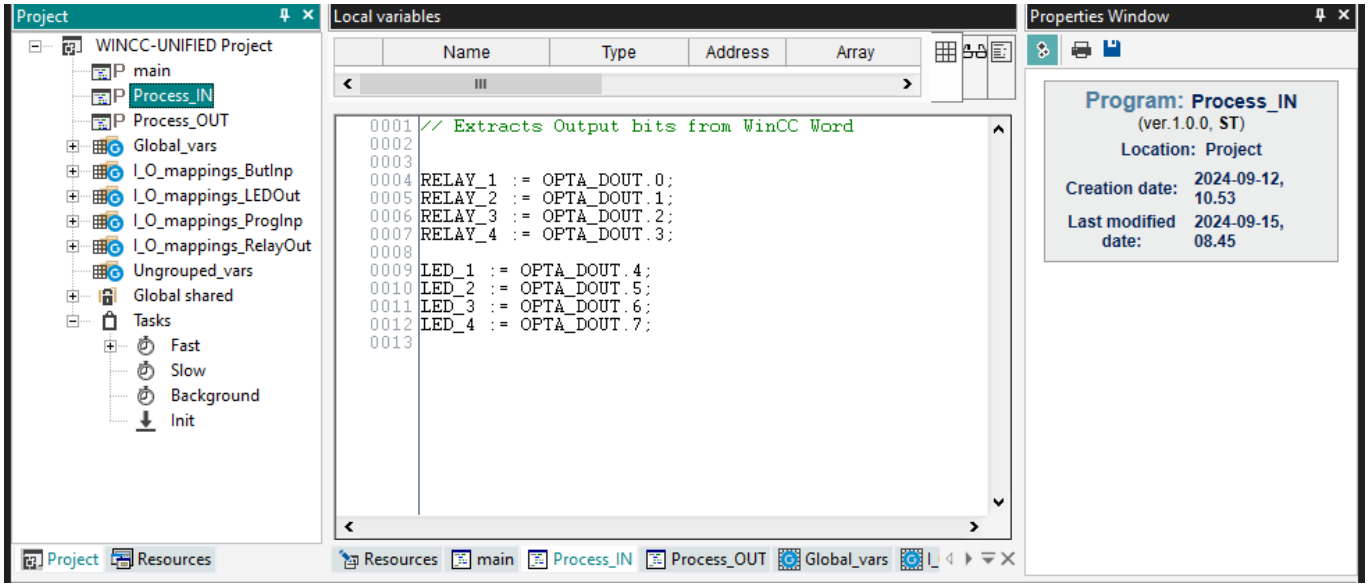


1. Right click to the first element into the Project Tab and, in the context menu, select **Add->New program**.

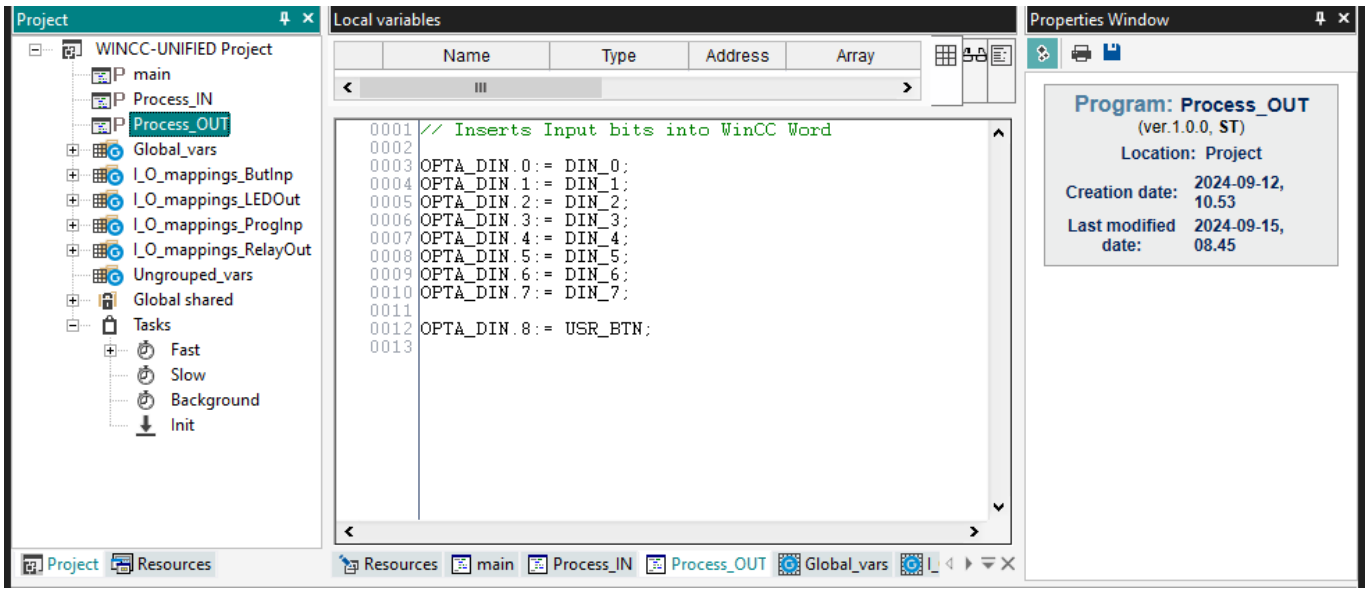
2. Create **two** ST Programs and call them **Process_IN** and **Process_OUT**. Assign them to the Fast Task.



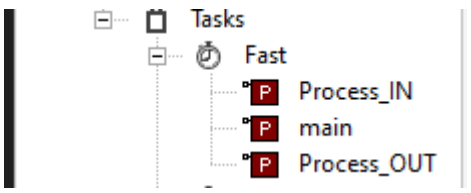
3. Fill **Process_IN**.



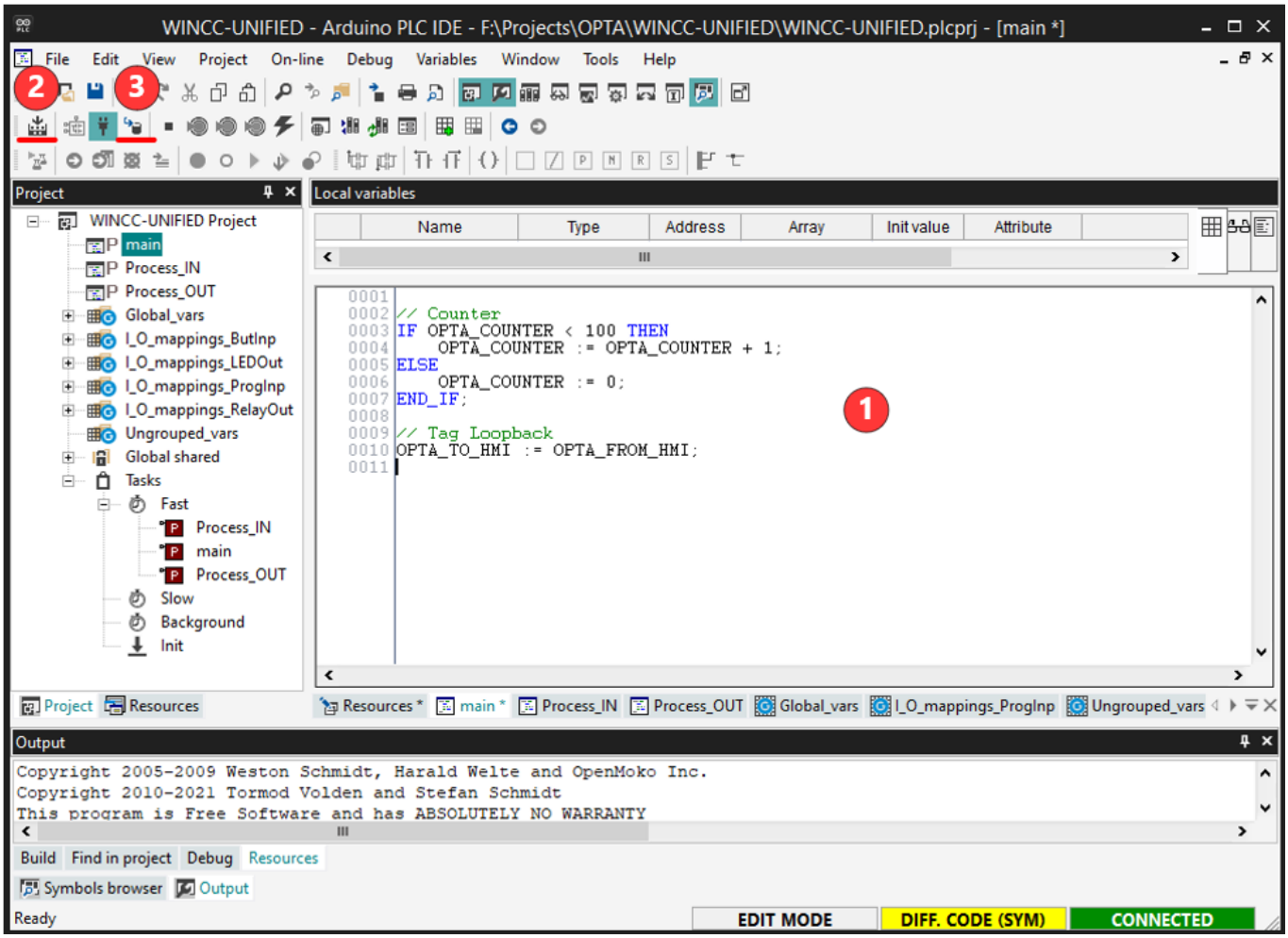
4. Fill **Process_OUT**.



5. Expand the **Fast Task** and arrange the programs as in figure.



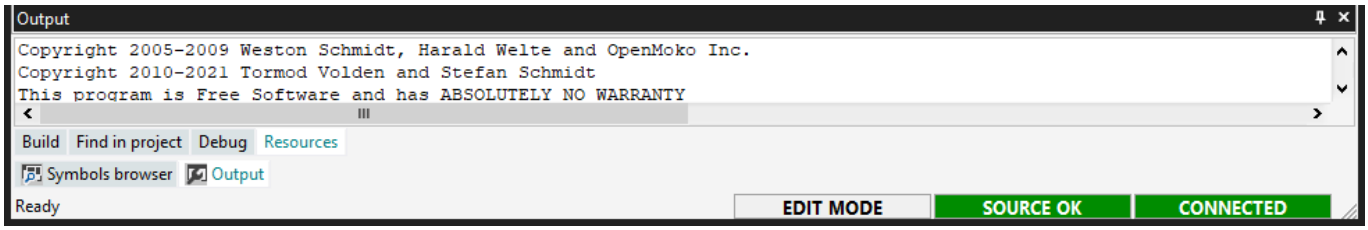
OPTA.5 – Preparing the main program



1. Fill the main program

2. Save and compile the project

3. Download it into the PLC



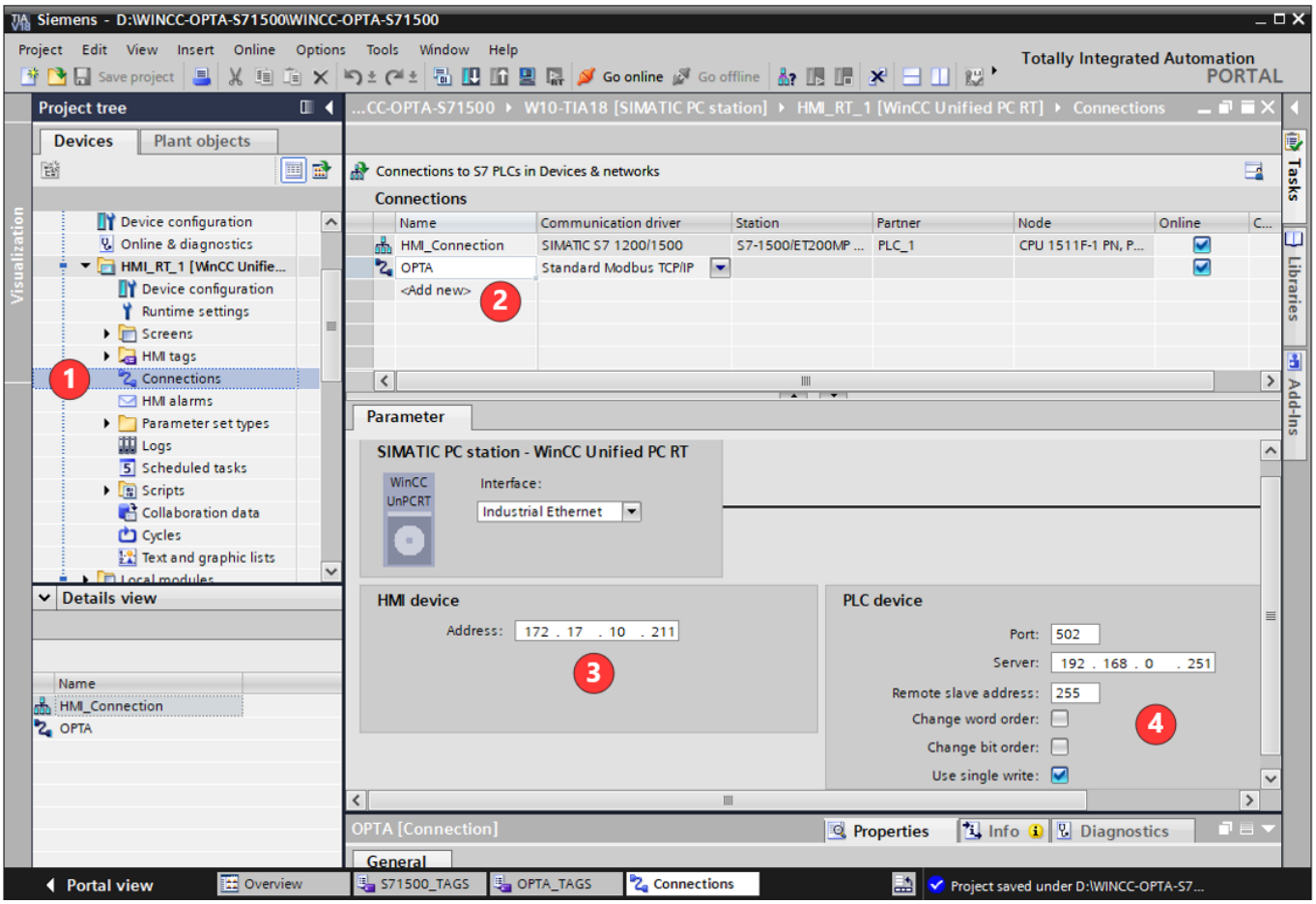
Now OPTA is ready.

Phase 2: Preparing WinCC Unified RT

To prepare WinCC, these steps are needed.

1. Creating and parameterizing the OPTA connection.
2. Creating interface Tags.
3. Creating the OPTA graphic page (called Screen into WinCC world)
4. Attaching Tags to the graphic objects

WinCC.1 – Creating the OPTA connection



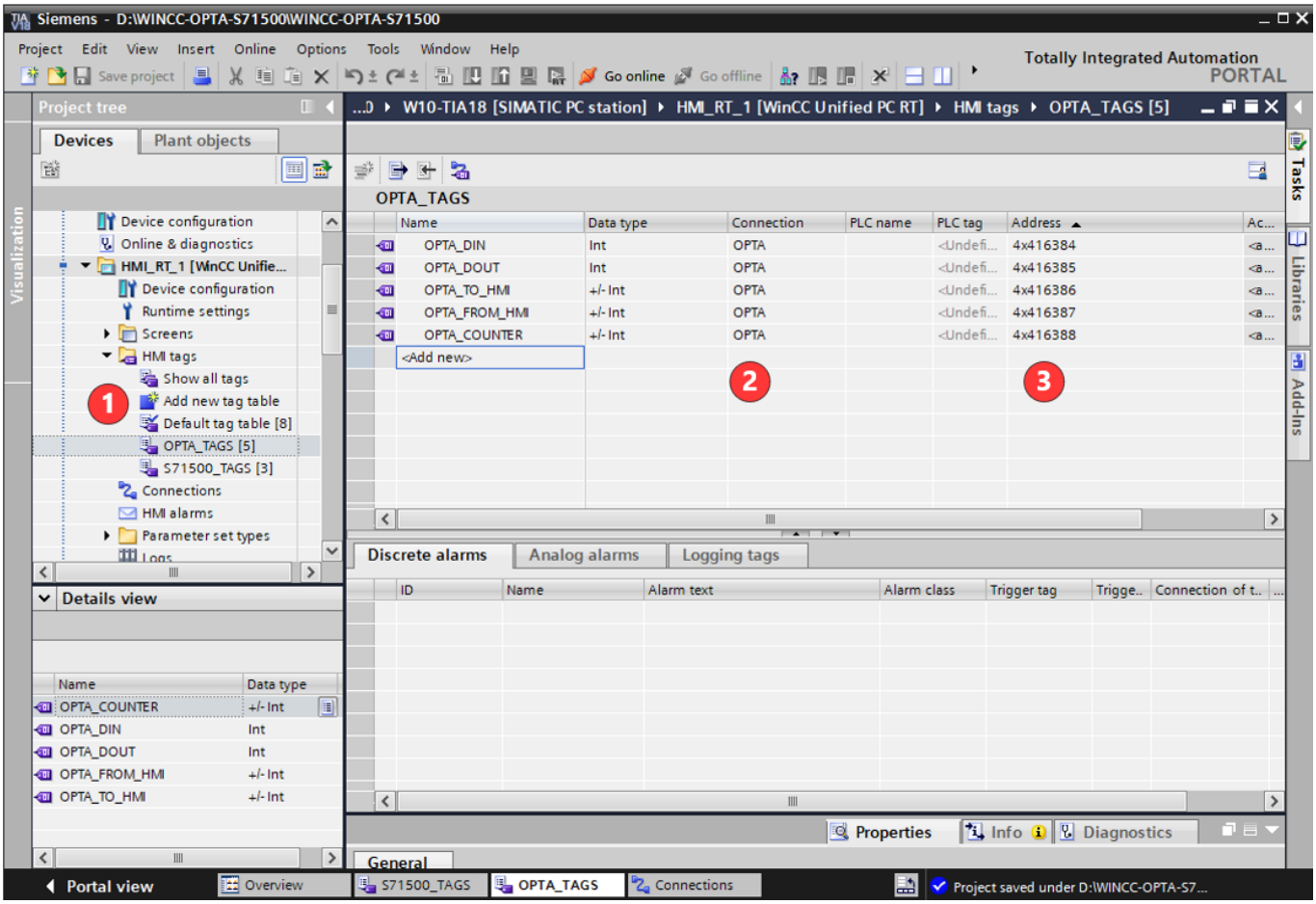
1. Select **Connections** under the WinCC project

2. Double click to **<add new>, r**ename the new connection to OPTA (only for cleanliness) and select **Standard Modbus TCP/IP** as driver.

3. Set the HMI IP Address.

4. Set OPTA IP address and Remote slave address as in figure. Leave checked the item Use single write. Leave 502 as port number.

WinCC.2 – Creating the interface Tags



1. Expand HMI Tags and double click to **Add new tag table**, rename the tags group to OPTA_TAGS (only for cleanliness)

2. Insert the tags and set their Data type and the connection.

3. Set the Tags addresses following this scheme

OPTA

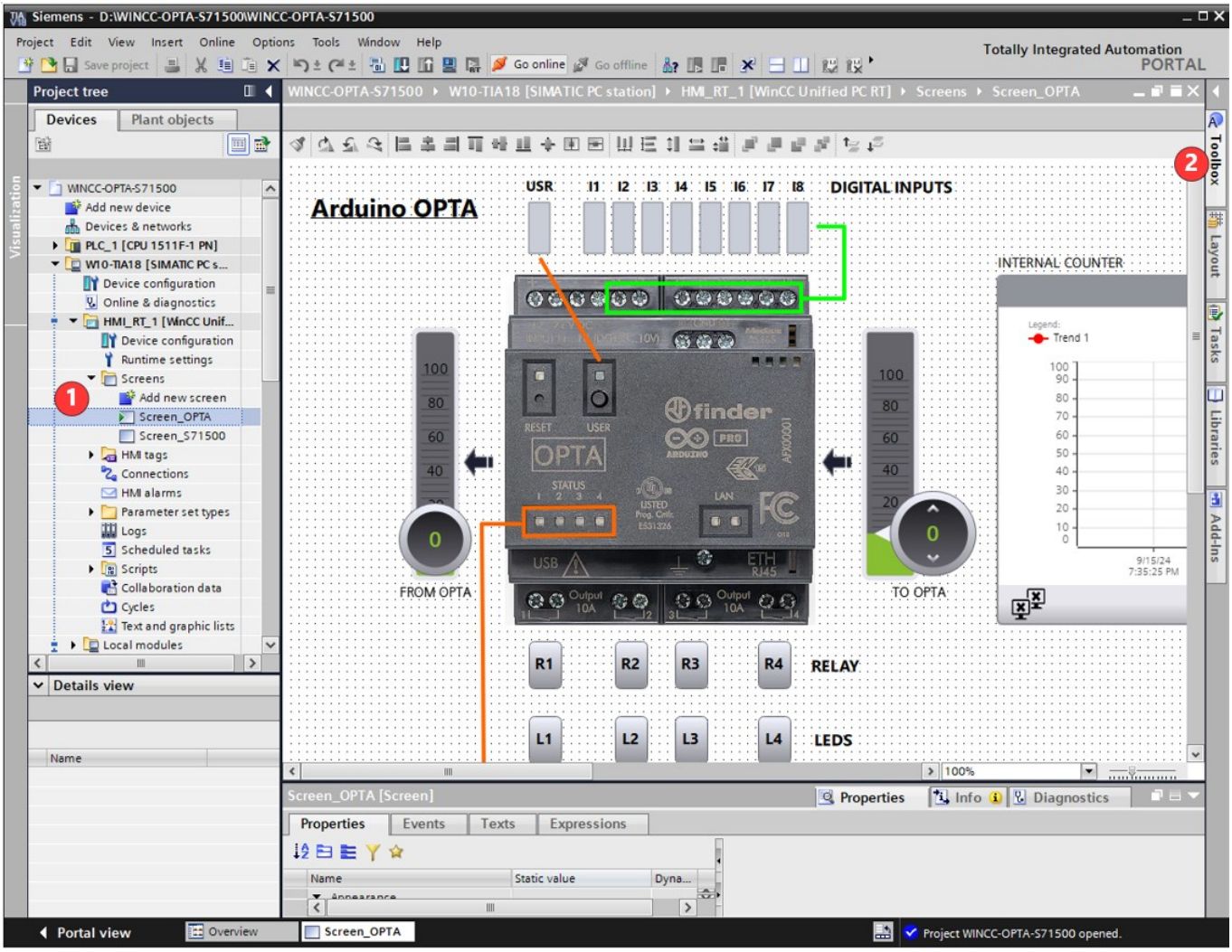
#	Address (dec)	Address (hex)	Name
1	16384	0x4000	OPTA_DIN
2	16385	0x4001	OPTA_DOUT
3	16386	0x4002	OPTA_TO_HMI
4	16387	0x4003	OPTA_FROM_HMI
5	16388	0x4004	OPTA_COUNTER

WinCC

Name	Data type	Address
OPTA_DIN	Int	4x416384
OPTA_DOUT	Int	4x416385
OPTA_TO_HMI	+/- Int	4x416386
OPTA_FROM_HMI	+/- Int	4x416387
OPTA_COUNTER	+/- Int	4x416388

16384 -> 4x416384

WinCC.3 – Creating the OPTA Screen



1. Expand **Screens** and double click to **Add new screen**, rename the page to Screen_OPTA (only for cleanliness).

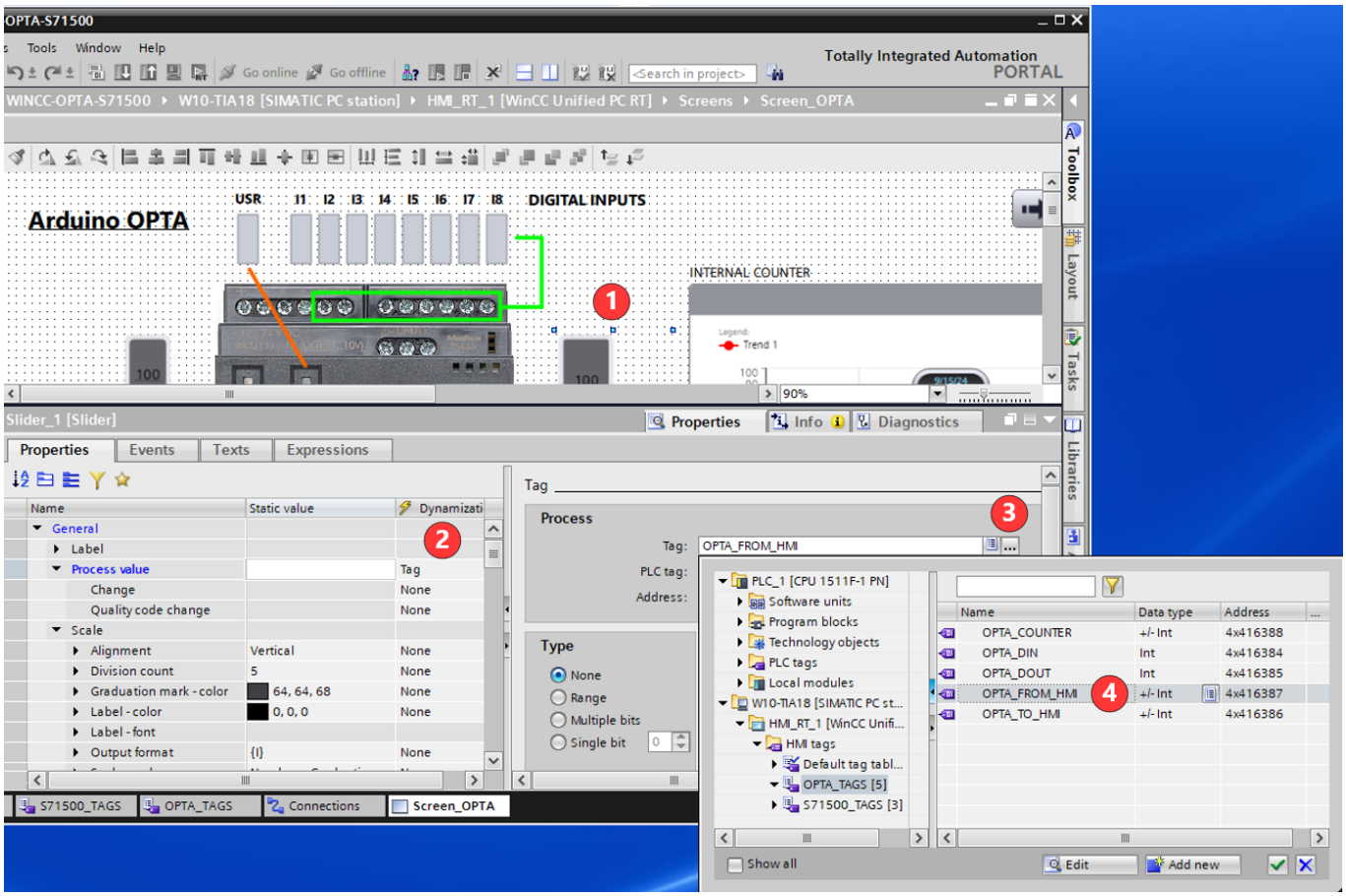
2. Expand the Toolbox page and drag the controls into the screen.

Explaining in detail the use of WinCC controls is beyond the scope of this tutorial, moreover this activity **does not change at all** for PLCs of different brands.

However, we will see a couple of examples to understand how to select the Tags correctly.

WinCC.4 – Attaching Tags to the controls

Example 1: parametrizing a Slider

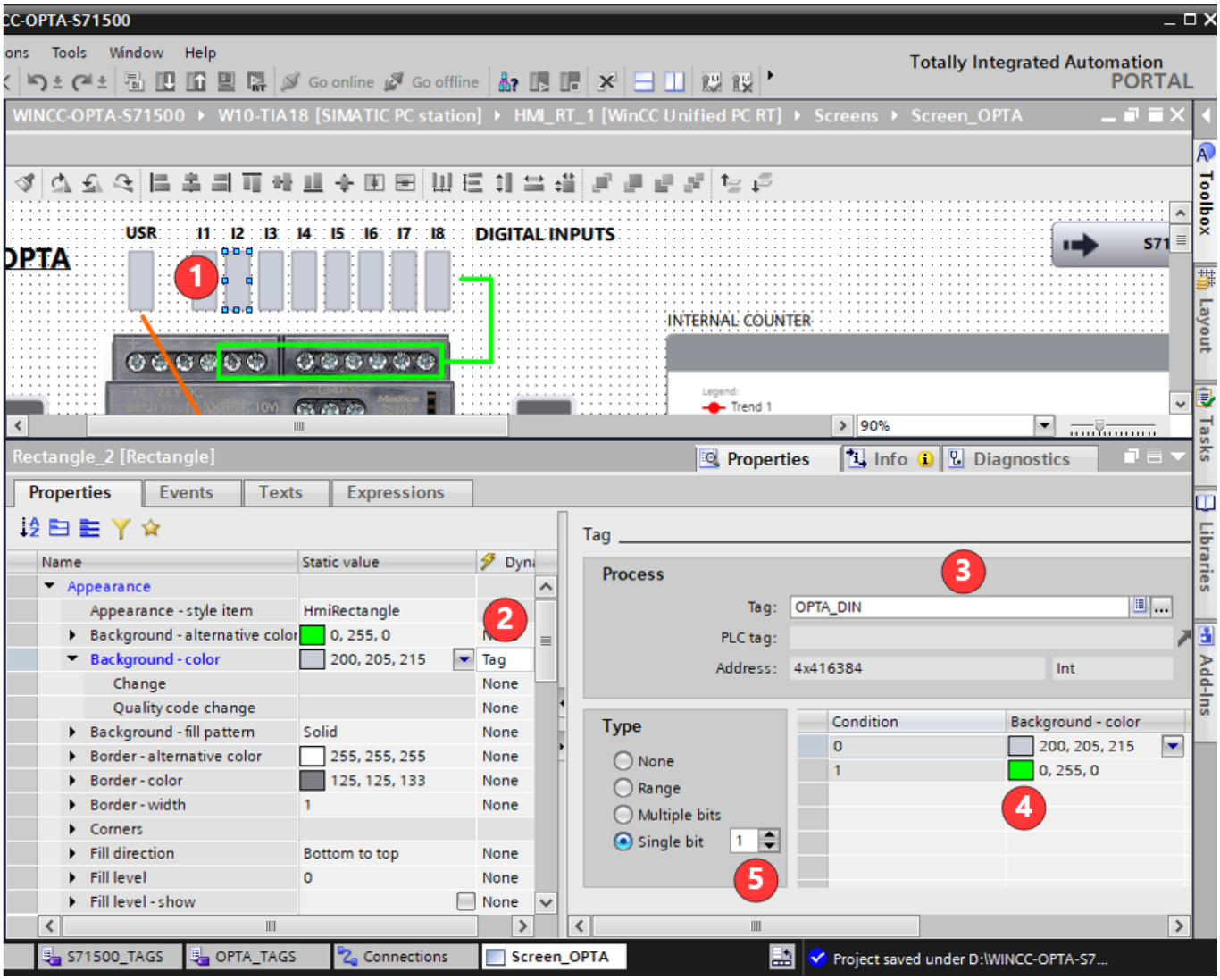


1. Insert a Slider, right click on it and select **Properties** in the context menu.

2. Edit the **Dinamization** column of the **Process value** property selecting **Tag**.

3. Click on […], navigate through **HMI Tags->OPTA TAGS** and select the Tag you want to link.

Example 2: parametrizing a LED



1. Insert a Rectangle, right click on it and select **Properties** in the context menu.

2. Edit the **Dinamization** column of the **Background - color** property selecting **Tag**.

3. Set the Tag as seen in the previous example.

4. Set the colors in accord to 0 and 1 values.

5. The tag that we used is a Word, so set the bit into the word (first bit is 0, second bit is 1, and so on).

Go Live!

There are two pages, this is what you will see: